Nonlinear multiscale transformations: From synchronization to error control

F. Arandiga and R. Donat

Dept. Matematica Aplicada, University of Valencia, Spain.
arandiga@uv.es donat@uv.es

Abstract

Data-dependent interpolatory techniques can be used in the reconstruction step of a multiresolution "*à la Harten*". These interpolatory techniques lead to nonlinear multiresolution schemes. When dealing with nonlinear algorithms, the issue of the stability needs to be carefully considered. In this paper we analyze and compare several strategies for image compression and their ability to effectively control the global error due to compression.

1 Introduction

Multiscale transformations are being used in recent times in the first step of transform coding algorithms for image compression. Ideally, a multiscale transformation allows for an *efficient* representation of the image data, which is then processed using a (non-reversible) quantizer and passed on to the encoder which produces the final compressed set of data which is ready to be transmitted or stored. Compression is indeed achieved during the second and third steps: the quantization and the encoding of the transformed set of discrete data.

It is quite clear that the properties of the multiscale transformation are most important in the overall performance of the transform coding algorithm. Until recently, the multiscale transformations used for image compression were always based on linear filter banks, however, the nonlinear alternative has been explored lately by various authors from different points of view, and preliminary results show the alternative to be very promising [12, 8, 6, 2, 3]. The key question when using, or even designing, a nonlinear multiscale transformation is that of stability. In order for such transformations to be useful tools in image coding, it is absolutely necessary to keep a tight control on the effect of quantization errors in the decoding process.

In this paper we examine the question of stability for nonlinear multiscale transformations within Harten's framework for multiresolution [14, 15]. Harten's framework is broad enough to include all classical wavelet transformations as particular cases (just as it happens in the Lifting framework of W. Sweldens [17], developed slightly later in time but independently), however the design of the multiscale transformation is done directly on the spatial domain.

The building blocks of Harten's multiresolution framework are two operators that connect adjacent resolution levels. The *Decimation* (or also, *Restriction*) operator is a linear operator which acts as a low-pass filter, extracting low-resolution information from a discrete data set. The *Prediction* operator (also *Projection*) uses low-resolution data to predict discrete data at a higher resolution level. It is precisely the *design* of this operator what distinguishes Harten's framework from all other multiresolution frameworks. The prediction operator is based on a *consistent Reconstruction* technique, and this opens up a tremendous number of possibilities in the design of multiresolution schemes. The use of the reconstruction process as a design tool makes it, conceptually, a simple matter to introduce adaptivity into the multiscale transformation; we only need to make the reconstruction process data-dependent [5, 4, 14].

This paper is organized as follows. In Section 2 we recall the so-called cell-average framework, an appropriate setting for image compression, and describe a class of nonlinear prediction operators obtained by mean-average interpolation [10, 14, 15]. In Section 3 we examine the question of stability for nonlinear multiscale transformations and relate it to the *synchronization* of the data-dependent choices made in the encoder and the decoder. We also include a set of numerical experiments that illustrate he performance of several nonlinear multiscale transformations.

2 Multiscale transformations in the cell-average setting

Harten's general framework for multiresolution [15] relies on two operators, Decimation and Prediction, that define the basic interscale relations. These operators act on finite dimensional linear vector spaces, V^j, that represent the different resolution levels (j increasing implies more resolution)

$$\text{(a) } D^j : V^j \to V^{j-1}, \qquad \text{(b) } P_j : V^{j-1} \to V^j, \tag{2.1}$$

and must satisfy two requirements of algebraic nature; D^j needs to be a *linear* operator and $D^j P_j = I_{V^{j-1}}$, i.e., the identity operator on the lower resolution level represented by V^{j-1}. For all practical purposes, V^j can be considered as spaces of finite dimensional sequences.

Using these two operators, a vector (i.e., a discrete sequence) $v^j \in V^j$ can be decomposed and reassembled as follows

$$\text{(a)} \quad \begin{array}{ccccc} v^j & \to & v^{j-1} & = & D^j v^j \\ & \searrow & e^j & = & v^j - P_j v^{j-1} \end{array}, \qquad \text{(b)} \quad v^j = P_j v^{j-1} + e^j \tag{2.2}$$

where e^j represents the error in trying to predict the jth level data, v^j, from the low resolution data $v^{j-1} = D^j v^j$, using the prediction operator P_j .

In the cell-average setting, the discrete data are interpreted as the cell-averages of a function on an underlying grid, which determines the level of resolution of the given data. The one dimensional case, in which one considers a set of nested dyadic grids on the interval $[0, 1]$, $\{X^j\}$, $j \geq 0$ of size $h_j = 2^{-j} h_0$,

$$X^j = \{x_i^j\} \qquad x_i^j = i \cdot h_j, \quad i = 0, \ldots, N_j \qquad N_j \cdot h_j = 1 \tag{2.3}$$

is the easiest one to describe, and it is also directly applicable to two-dimensional (2D) data via tensor product [2, 3] (the cell-average framework in several dimensions and non-tensor product (unstructured) grids is considered in e.g. [1]).

In this simple one-dimensional setting, the cell-average framework is characterized by the following decimation operator D^j

$$(D^j v^j)_i = \frac{1}{2}(v^j_{2i-1} + v^j_{2i}), \qquad 1 \le i \le N_{j-1}, \tag{2.4}$$

where N_j is the number of equally spaced intervals on X^j, the grid on $[0,1]$ that represents the jth resolution level. The *consistency* requirement for the prediction operator, i.e., $D^j P_j = I_{V^{j-1}}$ which is the only necessary requirement for the prediction in Harten's framework, becomes then

$$(P_j v^{j-1})_{2i-1} + (P_j v^{j-1})_{2i} = 2v^{j-1}_i. \tag{2.5}$$

Observe that (2.4) and (2.5) imply that

$$(P_j v^{j-1})_{2i} + (P_j v^{j-1})_{2i-1} = v^j_{2i} + v^j_{2i-1}.$$

Hence

$$e^j_{2i-1} = v^j_{2i-1} - (P_j v^{j-1})_{2i-1} = (P_j v^{j-1})_{2i} - v^j_{2i} = -e^j_{2i}.$$

Therefore the prediction errors at even and odd grid points on the jth level in (2.2) are not independent. By considering only the prediction errors at (for example) the odd points of the grid X^j, one immediately gets a one-to-one correspondence between the sets $\{v^j_i\}_{i=1}^{N_j} \leftrightarrow \{\{v^{j-1}_i\}_{i=1}^{N_{j-1}}, \{d^j_i\}_{i=1}^{N_{j-1}}\}$, with $d^j_i = e^j_{2i-1}$ and $v^{j-1} = D^j v^j$. The one-dimensional multiscale transformation and its inverse can be written as follows,

$$v^L \longrightarrow Mv^L = (v^0, d^1, \ldots, d^L) \quad \left\{ \begin{array}{l} \text{For } j = L, \ldots, 1 \\ \quad \text{For } i = 1, \ldots, N_{j-1} \\ \qquad v^{j-1}_i = (v^j_{2i} + v^j_{2i-1})/2 \\ \qquad d^j_i = v^j_{2i-1} - (P_j v^{j-1})_{2i-1} \end{array} \right\}, \tag{2.6}$$

$$v_d = (v^0, d^1, \ldots, d^L) \longrightarrow M^{-1} v_d \quad \left\{ \begin{array}{l} \text{For } j = 1, \ldots, L \\ \quad \text{For } i = 1, \ldots, N_{j-1} \\ \qquad v^j_{2i-1} = (P_j v^{j-1})_{2i-1} + d^j_i \\ \qquad v^j_{2i} = 2v^{j-1}_i - v^j_{2i-1} \end{array} \right\}. \tag{2.7}$$

Observe that since $d^j_i = e^j_{2i-1} = -e^j_{2i}$, the consistency relation (2.5) implies that the computation of v^j_{2i} in (2.7) is equivalent to

$$v^j_{2i} = 2v^{j-1}_i - v^j_{2i-1} = (P_j v^{j-1})_{2i} - d^j_i = (P_j v^{j-1})_{2i} + e^j_{2i}. \tag{2.8}$$

Therefore (2.6) and (2.7) are just the repeated application of the decomposition and reassembling specified in (2.2)(a) and (2.2)(b). Thus (2.6) defines a multiscale transformation and (2.7) is the inverse transformation, whether or not the prediction operator is linear.

Next, we follow [4, 14, 15] to describe a class of *linear* prediction operators that leads to the $(1, M)$ branch of the Cohen-Daubechies-Feauveau family [7], which is biorthogonal

to the box function [11, 15]. This class is also considered in [6] within the lifting framework, where it is described as a particular case of Donoho's average interpolation [9].

Given an integer $s \geq 1$, for each $1 \leq i \leq N_{j-1}$ we construct a polynomial, $p_i(x)$, of degree 2s such that

$$\frac{1}{h_{j-1}} \int_{x_{i+l-1}^{j-1}}^{x_{i+l}^{j-1}} p_i(x)dx = v_{i+l}^{j-1}, \qquad \text{for } l = -s, \ldots, s. \tag{2.9}$$

There are various ways to prove that $p_i(x)$ in (2.9) always exists and it is uniquely defined by the $2s+1$ conditions in (2.9) [1, 9, 14]. Then we define

$$(P_j v^{j-1})_{2i} = \frac{1}{h_j} \int_{x_{2i-1}^{j}}^{x_{2i}^{j}} p_i(x)dx, \qquad (P_j v^{j-1})_{2i-1} = \frac{1}{h_j} \int_{x_{2i-2}^{j}}^{x_{2i-1}^{j}} p_i(x)dx. \tag{2.10}$$

The prediction operator defined by (2.10) is data-independent, hence linear, and it clearly satisfies the consistency relation (2.5). It can be shown that the multiscale transformations (2.6) and (2.7) for this class of prediction operators turns out to be the $(1, M = 2s+1)$ branch of the Cohen-Daubechies-Feauveau family.

A nonlinear prediction operator is obtained if we construct $p_i(x)$ in a data-dependent way. An example of nonlinear multiresolution transformation constructed in this fashion is considered in [14, 4, 2], where a nonlinear ENO-type technique (Essentially Non Oscillatory, see [16]) is used to construct $p_i(x)$. The key idea, which is in essence common to the approach used in designing nonlinear filter banks, is to avoid using data across an edge for the prediction step.

The ENO nonlinear technique is better described if we associate to each polynomial piece $p_i(x)$ a *stencil*, $\mathcal{S}_i$, which is the set of indices of the values used to define $p_i(x)$. In the linear case $\mathcal{S}_i = \{i-s, \ldots, i+s\}$; the stencil is *independent* of the data set $\{v^{j-1}\}$ and, as a consequence, P_j is a linear operator. In the ENO technique described in [16], the selection of stencil is made in a data-dependent way using the divided differences of the data as a measure of its smoothness. Large divided differences occur when considering data across an edge, while divided (or undivided) differences of data on smoother regions tend to be smaller in size.

The information contained in the divided differences is then used to decide what is $\mathcal{S}_i$ for each i, with the only restriction that $i \in \mathcal{S}_i$ (to satisfy the consistency requirement (2.5)). We follow [4] and consider all polynomial pieces of the same degree. In our case $\#\mathcal{S}_i = 2s$, but in principle one could decide to lower the degree of $p_i(x)$, or that of some of its neighbours, whenever an edge-detection mechanism finds an edge at the ith interval. By lowering the degree of some polynomial pieces close to an edge, one can avoid crossing the edge in the prediction step, as much as possible. This option is closely related to the nonlinear multiscale transformation considered in [6] (within the Lifting framework), where the nonlinearity comes in from adaptively choosing from the $(1, M)$ family of linear filters.

Once $\mathcal{S}_i$ is determined ($i \in \mathcal{S}_i$), $p_i(x)$ can be uniquely determined when degree $p_i(x) =$

$\#\mathcal{S}_i$ [1] so that

$$\frac{1}{h_{j-1}} \int_{x_{m-1}^{j-1}}^{x_m^{j-1}} p_i(x)dx = v_m^{j-1} \qquad \text{for } m \in \mathcal{S}_i, \tag{2.11}$$

and the prediction operator is then defined by (2.10).

One can be slightly more 'sophisticated' in the design of the polynomial pieces. The *Subcell Resolution* technique [4, 13] allows to account for discontinuities within a cell as follows. If an edge is detected in the ith cell, the polynomial piece $p_i(x)$ is discarded and substituted by its left and right neighbours, $p_{i+1}(x)$ and $p_{i-1}(x)$, assuming that their respective stencils do not intersect, i.e. $\mathcal{S}_{i-1} \bigcap \mathcal{S}_{i+1} = \emptyset$. At a true one-dimensional edge (a jump) on the ith cell, the function

$$F(y) = \frac{1}{h_j} \int_{x_{2i-1}^j}^{y} p_{i-1}(x)dx + \frac{1}{h_j} \int_{y}^{x_{2i}^j} p_{i+1}(x)dx$$

will have a zero on the ith cell [13], say η, and the location of η is used to substitute the polynomial piece $p_i(x)$ by the discontinuous piecewise polynomial function

$$q_i(x) = \begin{cases} p_{i-1}(x) & x \leq \eta, \\ p_{i+1}(x) & x > \eta. \end{cases} \tag{2.12}$$

The prediction operator is again defined by (2.10) at *nonsingular* cells (cells in which no edge has been detected), while at the *singular* cell

$$(P_j v^{j-1})_{2i} = \frac{1}{h_j} \int_{x_{2i-1}^j}^{x_{2i}^j} q_i(x)dx, \qquad (P_j v^{j-1})_{2i-1} = \frac{1}{h_j} \int_{x_{2i-2}^j}^{x_{2i-1}^j} q_i(x)dx.$$

In practice it is unnecessary to compute explicitly the value of η; only its location with respect with x_{2i-1}^j is needed, which can be found by a sign check. We refer the reader to [4] (and references therein) for specific details on this technique, in particular on the detection mechanism, and on its performance.

3 The question of stability: Error control versus synchronization, with numerical examples

Lossy coding schemes introduce errors into the transform coefficients, and it becomes crucial that the nonlinearities do not unduly amplify these errors. In lossy compression the decoder only has the quantized detail coefficients. If we use a nonlinear prediction operator (whether it is constructed as described in the previous section or based on locally adapted filters, as in [6] within the Lifting framework), the quantization errors in coarse scales could cascade across the scale ladder and cause a series of incorrect choices (either on the filters or on the stencils) leading to serious reconstruction errors.

To avoid incorrect choices in the prediction step, whether within Harten's or the Lifting framework, one would need to send side information on which filter was used (Lifting) or what was the interpolatory stencil (Harten's). This is clearly inappropriate when trying to design a compression scheme. One way to avoid storing (and sending) side information is to somehow *synchronize* the nonlinear prediction operators in the encoder

and the decoder, so as to ensure that at a given spatial location on a given scale, the prediction operator will select the same stencil (filter bank), both in the encoding and the decoding steps.

Within the Lifting framework, synchronization is achieved in [6] by changing the typical Split-Predict-Update steps to Split-Update-Predict. In doing so, it is possible to base the *choice* of predictor directly on already 'quantized data', thus synchronizing the nonlinear decisions made by the encoder and the decoder.

Within Harten's framework, synchronization is just a consequence of a strategy that is designed to *fully control* the compression error. Because the main design tool in Harten's framework for multiresolution is a reconstruction technique, and because A. Harten had already worked with nonlinear reconstruction techniques in the context of the numerical simulation for hyperbolic conservation laws, so-called *Error-Control* (EC) strategies can be found already in the early papers of Harten on multiresolution [14].

Harten's mechanism to control the global accumulated error is based on a modification of the direct multiscale transformation, M, that ensures a prescribed tolerance on the global prediction errors (explicit error bounds can be found in [4, 13]). The modified transformation incorporates the quantizer to the direct multiscale transformation in such a way that the prediction operator in the encoder also acts on already 'quantized' data, hence synchronization is achieved because the nonlinear prediction operators both in M and M^{-1} work on the *same* set of discrete data at each resolution level.

To illustrate the effect of the different techniques, we take a particular nonlinear prediction operator, a third order ENO reconstruction technique with Subcell Resolution, as described in last section. We denote by M_{SR} the multiscale transformation (2.6), while M_{SR}^{M} denotes the EC modified transform as described in [2, 4], and M_{SR}^{S} a multiscale transformation in which only synchronization is enforced, as proposed in [6]. The quantization step is carried out as follows:

$$\mathbf{qu}(d^j) = 2\epsilon_j \text{round}\left[d^j/(2\epsilon_j)\right]$$

and it is incorporated to the direct transformation in M_{SR}^{M} and M_{SR}^{S} (see [2, 6] for specific details), while in M_{SR} it is applied to the scale coefficients obtained after the transformation. In the numerical tests we report, we take $\epsilon_L = 8$ with $L = 4$ and $\epsilon_j = \epsilon_{j+1}/2$.

We consider two different images: the familiar image of Lena as an example of a 'real' image, and a purely geometrical image, to which texture has been added, as in [6].

After the direct transformation (plus the quantization step) has taken place, a lossless Lempel-Ziv compression algorithm is applied to reduce the size of the transformed image, then a *compression ratio* is computed as the number of bits of the compressed representation over the number of bits of the original image. To recover the original image, we undo the lossless compression and transform back using (2.7) in all three cases. The full compression algorithm is identified in each case by an acronym, 'ST' for M_{SR}, 'EC' for M_{SR}^{M} and 'SYNC' for M_{SR}^{S}.

In Tables 1 and 2 we compile a number of quantities that measure the 'quality' of the reconstructed image, and therefore the robustness and reliability of each multiresolution-based compression algorithm, the magnitude of the global compression error, measured

Method	$\|\cdot\|_\infty$	$\|\cdot\|_1$	$\|\cdot\|_2$	r_c	entropy
ST	258	5.71	9.08	11.3:1	.6449
SYNC	195	6.45	9.82	7.9:1	.8875
EC	25.4	4.47	5.73	9.7:1	.6850

TAB. 1. Geometrical image.

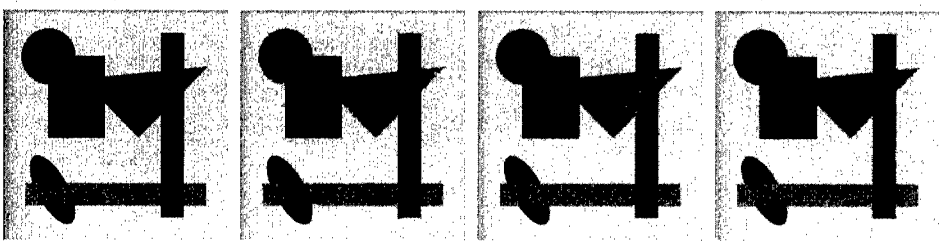

FIG. 1. Geometrical image: (a) original, (b) ST, (c) EC, (d) SYNC.

in various norms, the compression rate r_c and the entropy of the transformed image. The reconstructed images in both cases can be observed in Figures 1 and 2.

It can be clearly observed that the absence of any type of synchronization procedure can lead to a very poor reconstructed image. Synchronization *only* improves the quality, but is not as robust as the full EC mechanism, designed in this case to enforce a certain error bound in the 2-norm (as observed in Tables 1 and 2, the 2-norm of the global error is kept below $\epsilon_L = 8$). It is worth mentioning that the compression rate and the entropy of the compressed data are all very close, however the visual quality of the reconstructed image is significantly better for the EC compression algorithm.

Bibliography

1. R. Abgrall and A. Harten. Multiresolution representation in unstructured meshes. *SIAM J. Numer. Anal.* **35**, 2128–2146 (electronic), 1998.
2. S. Amat, F. Aràndiga, A. Cohen, and R. Donat. Tensor product multiresolution analysis with error control for compact image representation. Submitted to Signal Processing, 2000.
3. S. Amat, F. Aràndiga, A. Cohen, R. Donat, G. García, and M. Von Oehsen. Data compression with ENO schemes. *Applied and Computational Harmonic Analysis* **11**, 273–288, 2001.
4. F. Aràndiga and R. Donat. Nonlinear multi-scale decompositions: The approach of A. Harten. *Numer. Algorith.* **23**, 175–216, 2000.
5. F. Aràndiga, R. Donat, and A. Harten. Multiresolution based on weighted averages of the hat function II: Nonlinear reconstruction operators. *SIAM J. Sci. Comput.* **20**, 1053–1093, 1999.
6. R. L. Claypoole, G. Davis, W. Sweldens, and R. Baraniuk. Nonlinear wavelet transforms for image coding via lifting scheme. *submitted to IEEE Trans. on Image*

Method	$\|\cdot\|_\infty$	$\|\cdot\|_1$	$\|\cdot\|_2$	r_c	entropy
ST	318	5.66	10.59	8.8:1	.8261
SYNC	277	5.97	10.56	7.5:1	.9430
EC	26.4	3.59	4.84	8.2:1	.8704

TAB. 2. Lena.

FIG. 2. Lena: (a) original, (b) ST, (c) EC, (d) SYNC.

Processing., 1999.

7. A. Cohen, I. Daubechies, and J.C. Feauveau. Biorthogonal bases of compactly supported wavelets. *Comm. Pure Applied Math.* **45**, 485–560, 1992.
8. R. L. de Quieroz, D. A. Florêncio, and R. W. Schafer. Non-expansive pyramid for image coding using a non-linear filter bank. *IEEE Trans. Image Processing* **7**, 246–252, 1998.
9. D. L. Donoho. Interpolating wavelet transforms. Technical report, Department od Statistics, Stanford University, 1992.
10. D. L. Donoho and Thomas P.Y. Yu. Nonlinear pyramid transforms based on median-interpolation. *SIAM Journal on Mathematical Analysis* **31**, 1030–1061, 2000.
11. M. Guichaoua. *Analyses Multirésolution Biorthogonales associées à la Résolution d'Equations aux Dérivées Partielles.* PhD thesis, Ecole Superieure de Mecanique de Marseille, Universite de la Mediterranee Aix-Marseille II, 1999.
12. F.J. Hampson and J.C. Pesquet. A nonlinear subband decomposition with perfect reconstruction. In *Proce. IEEE Int. Conf. Acoust., Speech, and Signal Proc.*, 1996.
13. A. Harten. ENO schemes with subcell resolution. *J. Comput. Phys.* **83**, 148–184, 1989.
14. A. Harten. Discrete multiresolution analysis and generalized wavelets. *J. of Applied Num. Math.* **12**, 153–193, 1993.
15. A. Harten. Multiresolution representation of data: A general framework. *SIAM J. Numer. Anal.* **33**, 1205–1256, 1996.
16. A. Harten, B. Engquist, S. Osher, and S.R. Chakravarthy. Uniformly high-order accurate essentially nonoscillatory schemes. III. *J. Comput. Phys.*, **7** 1231–303, 1987.
17. W. Sweldens. The lifting scheme: a custom-design construction of biorthogonal wavelets. *Appl. Comput. Harmon. Anal.* **3**, 186–200, 1996.